

A Cognitive Architecture for Solving Ill-Defined Problems

Keith J. Holyoak

University of California at Los Angeles



Paul R. Thagard

Princeton University

for

**Contracting Officer's Representative
Judith M. Orasanu**

**Research and Advanced Concepts Office
Michael Drillings, Acting Director**

February 1995

19950424 059



DTIC QUALITY INSPECTED 8

**United States Army
Research Institute for the Behavioral and Social Sciences**

U.S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES

A Field Operating Agency Under the Jurisdiction
of the Deputy Chief of Staff for Personnel

EDGAR M. JOHNSON
Director

Research accomplished under contract
for the Department of the Army

University of California at Los Angeles

Technical review by

Joseph Psotka

Accession For	
NTIS	CRA&I <input checked="checked" type="checkbox"/>
DTIC	TAB <input type="checkbox"/>
Unannounced <input type="checkbox"/>	
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

NOTICES

DISTRIBUTION: This report has been cleared for release to the Defense Technical Information Center (DTIC) to comply with regulatory requirements. It has been given no primary distribution other than to DTIC and will be available only through DTIC or the National Technical Information Service (NTIS).

FINAL DISPOSITION: This report may be destroyed when it is no longer needed. Please do not return it to the U.S. Army Research Institute for the Behavioral and Social Sciences.

NOTE: The views, opinions, and findings in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other authorized documents.

REPORT DOCUMENTATION PAGE

Form Approved

OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE 1995, February	3. REPORT TYPE AND DATES COVERED Interim Sep 86 - Aug 87	
4. TITLE AND SUBTITLE A Cognitive Architecture for Solving Ill-Defined Problems			5. FUNDING NUMBERS MDA903-86-K-0297 601102A B74F 8513 C71	
6. AUTHOR(S) Holyoak, Keith J. (University of California, LA); Thagard, Paul R. (Princeton University)				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Department of Psychology University of California Los Angeles, CA 90024-1563			8. PERFORMING ORGANIZATION REPORT NUMBER --	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Research Institute for the Behavioral and Social Sciences ATTN: PERI-BR 5001 Eisenhower Avenue Alexandria, VA 22333-5600			10. SPONSORING / MONITORING AGENCY REPORT NUMBER ARI Research Note 95-24	
11. SUPPLEMENTARY NOTES Contracting Officer's Representative, Judith M. Orasanu.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE --	
13. ABSTRACT (Maximum 200 words) A computational theory of analogical mapping is described, based on a small set of constraints. The theory is embodied in a computer simulation that is applied to several examples, including psychological data on the mapping process.				
14. SUBJECT TERMS Analogy Constraint satisfaction Parallel processing			15. NUMBER OF PAGES 59	
			16. PRICE CODE ---	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT Unlimited.	

A COGNITIVE ARCHITECTURE FOR SOLVING ILL-DEFINED PROBLEMS

Executive Summary

During the first year of the project our work focussed on the following areas:

- (1) We developed a new theory of analogical mapping based on a small set of principles derived from data on human analogical thinking, coupled with computational considerations. The theory is described in the paper "Analogical mapping by constraint satisfaction: A computational theory," which has been submitted for publication and is included as the body of this annual report. The theory was implemented in a simulation written in COMMON LISP. The theory derives the optimal mapping between two complex analogs by computing the set of mappings between concepts and objects that best satisfies the constraints, using a parallel constraint-satisfaction algorithm. The program has been applied to over 10 examples, including detailed simulations of several sets of psychological data regarding the relative difficulty of various analogies. A graphics package was written to display the complex interactions between analogical components that yield the solutions to mapping problems.
- (2) A series of experiments was conducted to investigate the conditions under which people will spontaneously access analogies stored in memory. The results to date indicate that analogical access during problem solving is more likely to be triggered if the initial information was itself encoded in a problem-solving context, rather than having been encoded in a more declarative form.
- (3) A review paper entitled "Dimensions of analogy" was completed, reviewing alternative theories of analogy that have been proposed. This work was important in providing background for the development of a major revision and extension of our computational model of analogical problem solving.
- (4) Preliminary work was done to integrate the new model of the mapping process into our broader simulation of analogical problem solving. Continuation of this work will be a central focus of our research in the coming year.

Analogical Mapping by Constraint Satisfaction: A Computational Theory

Keith J. Holyoak

Paul Thagard

University of California, Los Angeles

Princeton University

Running head: Analogical Mapping

Address for proofs: Keith J. Holyoak

Department of Psychology

University of California

Los Angeles, CA 90024-1563

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Analogical mapping by constraint satisfaction: A computational theory		5. TYPE OF REPORT & PERIOD COVERED Interim (9/86 - 8/87)
7. AUTHOR(s) Keith J. Holyoak and Paul Thagard		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Psychology UCLA Los Angeles, CA 90024		8. CONTRACT OR GRANT NUMBER(s) MDA903-86-K-0297
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Research Institute 5001 Eisenhower Avenue Alexandria, Virginia 22333-5600		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBER 2Q161102B74F
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE Sept. 20, 1987
		13. NUMBER OF PAGES 51
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES Contracting Officer's Representative was Judith M. Orasanu		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) analogy, constraint satisfaction, parallel processing		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) A computational theory of analogical mapping is described, based on a small set of constraints. The theory is embodied in a computer simulation that is applied to several examples, including psychological data on the mapping process.		

Table of Contents

Abstract	2
Introduction	3
The Pragmatic Context of Analogical Mapping	4
The purpose of analogy	4
Subprocesses of analogy	5
Previous models of analogical mapping	8
A Theory on Analogical Mapping	10
The need for constraints	10
Five constraints on mapping	11
ACME: A cooperative algorithm for mapping	14
Applications of ACME	18
Problem analogies	18
Explanatory analogies	21
Story analogies	24
A formal isomorphism	27
Metaphor	28
General Discussion	29
Relationship of ACME to SME	29
Extensions of the theory	32
Footnotes	34
References	35
Figure Caption	39
Appendix	40

Abstract

We propose a computational theory of analogical mapping from a source analog to a target, based on five constraints. (1) *Logical compatibility* requires that elements of the source analog map only onto target elements of the same logical kind (constants to constants, *n*-place predicates to *n*-place predicates). (2) *Uniqueness* encourages each source element to map onto at most one element of the target. (3) *Relational consistency* leads any one hypothesis about mapped elements to support other hypothesized mappings suggested by the same relations. (4) *Semantic similarity* supports mapping hypotheses to the degree that mapped predicates have similar meanings. (5) *Role identity* restricts possible mappings to those between elements that play identical roles in high-level parts of the two analogs (e.g., goal elements can map only to goal elements). The theory is implemented in a computer program called ACME (Analogical Constraint Mapping Engine), which represents constraints by means of a network of supporting and competing hypotheses regarding what elements to map. A cooperative algorithm for parallel constraint satisfaction identifies mapping hypotheses that collectively represent the overall mapping that best fits the interacting constraints. ACME has been applied to a range of examples including problem analogies, explanatory analogies, story analogies, formal analogies, and metaphors. ACME is sensitive to similarity information if it is available, and yet able to compute mappings between formally isomorphic analogs with few similar or identical elements. The theory is able to account for empirical findings regarding the impact of consistency and similarity on human processing of analogies.

Introduction

At the core of analogical thinking lies the process of *mapping*: the construction of orderly correspondences between the elements of a source analog and those of a target. Identifying an appropriate mapping is crucial in allowing useful transfer of knowledge. In this paper we provide a computational theory of analogical mapping, and describe a program that embodies an algorithm for mapping analogs in accord with the theory.

The theory/algorithm distinction we have adopted derives from the work of David Marr (1982), who outlined a general approach to the analysis of information-processing systems. Such systems, he claimed, can be analyzed at three different levels. The level of *computational theory* focuses on the goal of the computation and the logic of the strategy by which it is carried out. The level of *representation and algorithm* addresses the representations of inputs and outputs and the algorithm that transforms the former into the latter. The level of *hardware realization* is concerned with the physical realization of the algorithm. Marr argued that although each of these levels of analysis can be important for understanding information processing, and all are logically and causally related, the couplings between them are quite loose. Some phenomena are best explained at only one or two of them. Marr stressed that analysis at the computational level is often crucial: "...An algorithm is likely to be understood more readily by understanding the nature of the problem being solved than by examining the mechanism (and the hardware) in which it is embodied" (1982, p. 27).

Although Marr's formulation has often been favorably discussed by cognitive scientists, it has less often been emulated, especially for aspects of information processing that lie within the domain of general higher-level reasoning, as opposed to perception and language. In part this may reflect the fact that Marr illustrated his approach by applying it to low-level perceptual problems such as stereopsis, for which there is good reason to think that common computational

principles apply across individuals and indeed across species. In contrast, higher-level cognitive processes are typically influenced by complex forms of learning, and often involve important variations in strategies. Although in principle a computational theory could allow for strategic variations in task performance, in practice such variations make it difficult to identify overarching computational principles.

In this paper we will apply Marr's form of theoretical analysis to analogical mapping. Despite the enormous conceptual gulf that would seem to separate analogical mapping from stereopsis, there are in fact suggestive similarities between the two problems at both the computational and algorithmic levels. Our analysis of analogy is thus to a degree *by* analogy. We will begin by considering the information-processing goals of analogical reasoning. The central function of mapping within this overall process can be realized by specifying the computation in terms of a small number of fundamental constraints. We will show that these constraints can be embodied in a robust parallel algorithm for analogical mapping, but make no claims at the level of hardware implementation.

The Pragmatic Context of Analogical Mapping

The Purpose of Analogy

Analogy, and inference in general, must be understood pragmatically, taking into account the goals and purposes of the cognitive system (Holland, Holyoak, Nisbett, & Thagard, 1986; Holyoak, 1985). At first glance, there seems to be no single information-processing purpose of analogy. Analogies are sometimes used to allow transfer of problem-solving operators, as when a problem of destroying a tumor is solved by analogy to a method of repairing lightbulbs; to provide explanations of various kinds, as when the behavior of heat is explained in terms of water flow; to construct scientific theories, as when sound is understood in terms of the behavior of water

waves; to identify relationships between formal systems, as when we note that both addition of numbers and union of sets exhibit the properties of commutativity and associativity; and to provide metaphors, as when we hear that "Life's... a tale told by an idiot, full of sound and fury, signifying nothing." What common thread might run through these diverse examples?

One essential function of analogy is to allow the mental representation of a *source* analog to organize the analogizer's mental representation of a *target* analog. The source provides a framework for analysing the target, selectively emphasizing aspects of the target that are congruent with the source, and potentially allowing the formation of novel hypotheses about the target domain. At the heart of this process, common to all examples of analogy, is the computation of a mapping—a set of correspondences—between elements of the source and target representations. The mapping should be *consistent*: mapped elements should remain in correspondence across the various relationships with other elements in which they participate. Consistency of mapping is crucial to establishing a shared organization between the source and target, rather than a set of arbitrary associations. Such a shared organization allows an analogy to serve the pragmatic function of using knowledge of the source to help understand the target.

Subprocesses of Analogy

In order to formulate a theory of mapping, it is necessary to consider the relationship between mapping and other aspects of analogical thinking. The centrality of mapping is a point of general agreement among all theorists who have discussed the use of analogy, whether in problem solving (Carbonell, 1983, 1986; Gick & Holyoak, 1980), in explanation (Gentner, 1983), in theory formation (Darden, 1983; Thagard, in press b), in the analysis of formal systems (Hesse, 1966; Polya, 1973), or in metaphor and other literary uses (Black, 1962; Gentner, 1982; Holyoak, 1982; Miller, 1979). There has been less agreement, however, on the relationship between mapping and

other subprocesses of analogy, and on the related issue of whether a common set of principles governs mapping in different pragmatic contexts.

It is useful to decompose analogy into four major components: (1) the retrieval or selection of a plausibly useful source analog, (2) mapping, (3) analogical inference or transfer, and (4) subsequent learning. Here we will leave aside the important issue of learning in the aftermath of analogy use, and focus on mapping and the components that immediately surround it: selection and transfer. These three subprocesses must collectively pick out a plausibly useful source analog, identify elements of the source that should determine transfer to the target, and effect such transfer.

Is there in fact a general-purpose mapping component that operates in fundamentally the same way for different varieties of analogy, and if so what role does it play in this overall task? We can address this question indirectly, by examining the functions performed by the subprocesses of selection and transfer, and then considering what remains. Clearly, the selection component is crucial to the success of analogy. Spontaneous retrieval of a relevant source analog depends on the presence of similar elements in the source and target, including (in the case of problem analogs) similar constraints and goals (Brown, Kane, & Echols, 1986; Holyoak & Koh, 1987). In the absence of clear similarities, useful analogies are often missed (Gick & Holyoak, 1980); if misleading surface similarities are present, false analogies may be accessed and lead to negative transfer (Novick, 1986).

Once a possible source analog is retrieved spontaneously or provided by a teacher, further selection must be made of the aspects of the source relevant to the analogy. Analogies are virtually always used to serve some known purpose, and the purpose will guide selection. If, for example, one is simply asked to compare what is known about Nicaragua with what is known about Cuba, all elements of the two representations are relevant. But if one is asked to assess

likely political trends in Nicaragua based on analogy to Cuba, then only a subset of what is known about Cuba—roughly, facts that bear on the development of its political system—will be mapped. For example, it is relevant to consider the degree to which Nicaragua's Daniel Ortega resembles Cuba's Fidel Castro. In contrast, suppose one is asked to predict the suitability of Nicaragua for sugar-cane production, again based on analogy to Cuba. The subset of knowledge about the source that is likely to be mapped will be very different—the similarity of Nicaragua to Cuba in terms of temperature and rainfall will loom much larger when the question concerns agriculture rather than politics. In examples such as these, the selection process can use pragmatic knowledge about the purpose of the analogy to identify not only a relevant source analog, but also *which aspects* of the source need be mapped. Much of the work of identifying aspects of the source that will determine transfer to the target can be done prior to mapping, based on knowledge of the purpose of the analogy coupled with causal knowledge concerning the source.

Similarly, knowledge can be brought to bear on the transfer process *after* mapping has established correspondences between elements of the source and target. The mapping implicitly defines a set of inferences that could be made about the target, based on correspondences with predicates and objects in the source domain. Thus if predicate *P* and object *O* in the source map onto *P'* and *O'* in the target, and the proposition *P(O)* holds in the source, then the proposition *P'(O')* is a candidate inference about the target. Whether a candidate inference will in fact be seriously considered as a plausible hypothesis about the target will depend on such pragmatic factors as whether the inference is relevant to the analogizer's goals in using the analogy and whether the inference is consistent with what is already known about the target domain.

The central task of the mapping component, then, is to take as inputs a target analog and a plausibly relevant source, and to compute a set of correspondences between elements of the source and target that is likely to yield useful candidate inferences. We will argue that in addition to

being embedded in the overall pragmatic context of analogy use, the mapping component itself is guided in part by pragmatic and semantic constraints.

Previous Models of Analogical Mapping

Numerous models of analogical mapping have been proposed by researchers in cognitive psychology and artificial intelligence, and we will not attempt a thorough review here (see Hall, 1986; Thagard, in press a). Different models have tended to stress pragmatic, semantic, or syntactic factors that might be used in mapping; few models have considered how different types of constraints might interact. Some models have focused on the roles of high-level plans, goals, and functional knowledge in determining the most appropriate mapping (Burstein, 1986; Carbonell, 1983, 1986; Kedar-Kabelli, 1985). Other models, such as that of Winston (1980), have stressed the importance of predicate similarity and causal knowledge. Winston's model used a serial algorithm that involved exhaustively considering each possible mapping of elements between source and target. The algorithm proved computationally intractable when the number of elements became even modestly large. As a result, it was necessary to limit the number of possible mappings by imposing arbitrary semantic restrictions, such as only mapping story characters of the same gender. As a consequence, the mapping algorithm would necessarily miss some potentially interesting mappings. These computational problems encountered by Winston suggest the importance of exploring alternative parallel algorithms.

The strongest proponent of the role of syntactic knowledge in guiding mapping and transfer is Gentner (1983, forthcoming; Falkenhainer, Forbus, & Gentner, 1986). Gentner's "structure-mapping" theory distinguishes between "attributes", which are 1-place predicates with objects as arguments; "first-order relations", which are multi-place predicates with objects as arguments; and "higher-order relations", which are multi-place predicates with propositions as arguments.

On her view, higher-order relations are more likely to be transferred from the source to the target than are first-order relations; attributes are not transferred at all in analogies. The structure-mapping theory stresses the role of a *systematicity principle* in guiding mapping: mappings between identical higher-order relations constrain mappings between first-order relations, which in turn constrain object mappings. To implement these ideas, Falkenhainer et al. (1986) have developed a computer program called SME (Structure-Mapping Engine).

Relational structure undoubtedly plays a major role in mapping. Furthermore, empirical evidence supports the theory's prediction that systematicity should correlate with ease of mapping (Gentner & Toupin, 1986). Nevertheless, several pieces of evidence suggest that the mapping component is not solely guided by syntactic constraints based on logical form. The particular higher-order relations emphasized in tests of the structure-mapping theory, "cause" and "implies", clearly might be deemed important on pragmatic rather than purely syntactic grounds (Hesse, 1966; Winston, 1980). Empirical evidence also indicates that semantic similarity of predicates, including attributes, has a major impact on ease of mapping (Gentner & Toupin, 1986). Furthermore, attributes sometimes seem to be more important than relations in analogical transfer, both in solving problems (Holyoak, Junn, & Billman, 1984) and in interpreting simple metaphors (e.g., "Tom is a giraffe").

Despite these problems, Gentner's structure-mapping theory has been applied successfully to a wide range of analogies. We concur with the general notion that predicate-argument relations help to constrain mappings, although we will challenge Gentner's characterization of systematicity in terms of higher-order relations. Because of its stress on the generality of the mapping component, and the clarity of its formulation, the structure-mapping theory will be used as the major basis for comparative assessment of the theory proposed here.

A Theory of Analogical Mapping

The Need for Constraints

The goal of our theory is to account for analogical mappings between mental representations of complex, organized bodies of knowledge. The mapping component is of course only one piece of an overall processing system for analogical reasoning. In addition to a natural-language interface, we assume prior processes of analogical retrieval and selection that (a) propose a plausible source-target pair, and (b) may provide information about the degree of semantic similarity between pairs of source-target predicates. The similarity computation may be based on decomposition of meanings into identities and differences (Hesse, 1966; Tversky, 1977). For our present purposes, however, we simply assume that the mapping component can receive a numerical index of the degree of semantic similarity between two predicates. In general, our theory of mapping can be stated independently of any strong theory of similarity, memory retrieval, or of other subprocesses of analogical inference.

The theory to be proposed is compatible with any account of mental representation rich enough to distinguish (a) between predicates such as *dog* and constants such as *Fido*, and (b) between predicates with different numbers of arguments. For example, *cow* is a one-place predicate taking one argument, as in *cow (Bossy)*, whereas *loves* is a two-place predicate taking two arguments, as in *loves (John, Mary)*. The algorithm described below takes as input sentences in first-order predicate calculus, but the theory is not wedded to that particular formalism.

The fundamental problem of analogical mapping is how to find appropriate correspondences between two analogs. If the analogs each have m predicates and n constants, then there are $m!n!$ possible mappings from which to select. Thus a typical analogy between analogs with ten predicates and five constants each generates over four hundred million possible mappings. Efficient selection of the best mapping requires that some constraints be placed on what it might

be. This problem is similar to that of stereoscopic vision (Marr & Poggio, 1976). Stereopsis requires that points in two visual images, one from each eye, be appropriately paired; however, there is no *a priori* basis for uniquely deciding which point should go with which. Similarly, given representations of two complex analogs, there is no *a priori* basis for establishing a determinate set of correspondences between elements in the two analogs. In order to account for stereopsis, Marr and Poggio proposed several qualitative constraints on the visual system. These constraints lead to the emergence of a unique set of point-to-point pairings, with each pairing consisting of points in each image arising from the same spatial position in the environment. Our computational theory of analogical mapping will similarly consist of a set of constraints.

Five Constraints on Mapping

Our constraint-satisfaction theory of mapping is based on five constraints that help to identify useful mappings.

(1) *Logical compatibility* requires that elements of the source analog map only onto target elements of the same logical kind (constants to constants, *n*-place predicates to *n*-place predicates). Consider, for example, the analogy between the problems of using a laser to fuse a broken filament in a lightbulb and of using an X-ray to destroy a tumor in a patient's stomach (Holyoak & Koh, 1987). The logical-compatibility constraint would preclude consideration of a mapping between the one-place predicate *laser* and the two-place relation *destroy*.

(2) *Uniqueness* encourages each source element to map onto at most one element of the target, and no two source elements to be mapped to the same target element. This constraint favors mappings that yield isomorphisms.

(3) *Relational consistency* leads a hypothesis about mapped elements to support other hypothesized mappings suggested by the same relations. In the above example, consistency

implies that if *fuse* maps onto *destroy*, then *filament* should tend to map onto *tumor*. Relational consistency helps produce a mapping that can be used to generate inferences about the target.

(4) *Semantic similarity* supports mapping hypotheses to the degree that mapped predicates have similar meanings. In addition to empirical evidence that predicate similarity influences mapping (Gentner & Toupin, 1986; Ross, in press), the similarity constraint is clearly pragmatically useful. Similar elements are in general likely to serve similar functions, and highly similar mapped predicates will yield a rich set of candidate inferences (Hesse, 1966). In our example, similarity would favor the mapping hypothesis *laser-X-ray* over *laser-tumor*.

(5) *Role identity*, when applicable, restricts mapping hypotheses to those involving elements that play identical roles in the two analogs. The roles are based on pragmatically important divisions of the analogs into high-level parts. In the program described below, role identity only operates for problem analogies. Problems can be divided into the basic parts of initial state, goal state, solution constraints, and operators (Carbonell, 1983; Newell & Simon, 1972). Role identity requires that initial states map to initial states, goals to goals, and so on. In the above example, mappings involving *fuse* and *destroy* would be considered because each is part of the goal state of a problem.

These five constraints serve differing functions in the mapping process. Logical consistency and role identity serve to prune the space of mapping hypotheses to be considered; the other constraints serve to evaluate the resulting set of candidate hypotheses. Uniqueness and relational consistency are purely internal constraints on mapping, which determine the intrinsic "goodness" of an analogy. Semantic similarity plays a more heuristic role, making it easier to find mappings that conform to the norm of having similar elements having similar functions.

The above constraints on analogical mapping bear interesting similarities to the constraints on stereopsis proposed by Marr and Poggio (1976). Logical consistency and role identity have

no apparent analog in the visual domain, but semantic similarity serves a function much like that of physical similarity (points of equal brightness may match). Uniqueness constrains both analogy and stereopsis. Relational consistency roughly corresponds to the visual constraint that disparities between matched points tend to vary continuously, in that both constraints allow evidence favoring any individual match hypothesis to be propagated to related hypotheses.

These substantive similarities between constraints presumably reflect underlying commonalities between the computational tasks of mapping and matching. However, the differences between the tasks are at least as significant as the similarities. Most notably, there is good reason to think that the constraints on stereopsis are biologically determined and largely invariant across individuals. In contrast, the constraints on analogical mapping may be influenced by learning and be open to strategic variations involving the relative influence of different constraints. Individual differences in analogical reasoning may arise from several factors:

- (1) Only a subset of the mapping hypotheses consistent with the constraints may in fact be considered. The selection may be based on additional constraints or simply on limitations of processing capacity. For example, only mappings between similar predicates may ever be considered. Such a strategy would preclude finding potentially useful mappings between dissimilar elements.

- (2) Given the heuristic nature of the semantic similarity constraint, an intelligent strategy would be to enforce the constraint early in the mapping process, and then progressively relax it as the process proceeds. This strategy would allow the analogizer to derive the heuristic benefit of the similarity constraint in guiding the mapping process, but reduce the attendant danger of being "trapped" if the optimal mapping as defined by uniqueness and relational consistency in fact requires violation of the similarity constraint.

- (3) The use of role identity is particularly dependent on learning. Without clear role

divisions that can be imposed on the analogs, this constraint cannot be invoked. It follows, for example, that an analogizer who encodes problems in terms of an abstract part structure may be better able to solve problems by analogy than will one who lacks such a structure.

Although the simulation of analogical mapping described below does not directly model such strategic variations in mapping, it should be clear that the general theory provides a framework for incorporating such variations.

ACME: A Cooperative Algorithm for Mapping

What kind of representations and algorithms might be appropriate for computing analogical mappings on the basis of constraints? With respect to representation, our theory makes only the minimal assumption of distinguishing among constants and predicates of different numbers of arguments. Our algorithm for setting up possible mappings between analogs takes as input sets of sentences in first-order predicate calculus. We have no particular devotion to predicate calculus as a representation language (Thagard, 1984), but use it here because of its simplicity and familiarity. Other more complex representation languages should be amenable to similar treatment.

Our algorithm for evaluating mappings is suggested by Marr and Poggio's (1976) treatment of stereoscopic matching, which was based on a *cooperative* algorithm, "...so-called because of the way in which local operations appear to cooperate in forming global order in a well-regulated manner" (Marr, 1982, p. 122). A cooperative algorithm is a procedure for parallel satisfaction of a set of interacting constraints. In the Marr and Poggio algorithm, a network of nodes is established, in which each node represents a possible pair of matched points, and excitatory and inhibitory connections between nodes represent constraints. The network is then allowed to run in order to find a globally optimal set of match hypotheses.

More generally, Marr (1982) argued that cooperative methods capture two principles that appear to govern fluent information processing: (1) the principle of *graceful degradation*, according to which degrading the input data should allow computation of a partial answer, and (2) the principle of *least commitment*, which requires avoiding doing something that may later have to be undone. More recently, others have argued that cooperative methods may be applicable to human memory-retrieval and higher-level reasoning, in addition to perceptual tasks (Rumelhart, Smolensky, McClelland, & Hinton, 1986). Several properties of an information-processing task can provide cues that a cooperative algorithm may be appropriate. A cooperative algorithm for parallel constraint satisfaction is preferable to any serial decision procedure when: (a) a global decision is composed of a number of constituent decisions, (b) each constituent decision should be based on multiple constraints, (c) the outcome of the global decision could vary depending on the order in which constraints are applied and constituent decisions are made, and (d) there is no principled justification for preferring any particular ordering of constraints or of constituent decisions. (For a philosophical discussion of the importance of parallel computation, see Thagard 1986.)

Analogical mapping using constraints exhibits all of these features. Accordingly, we have formulated a cooperative algorithm for mapping analogies and implemented it in a COMMON LISP program called ACME (Analogical Constraint Mapping Engine). Each possible mapping hypothesis about a possible pairing of a predicate or constant from the source with a corresponding element of the target is assigned to a node or *unit*. Each unit has an *activation level* ranging from -1 to 1, which indicates the plausibility of the corresponding hypothesis. Activation of 1 indicates maximal plausibility and activation of -1 indicates maximal implausibility. Inferential dependencies between mapping hypotheses are represented by *weights* on *links* between units. Supporting evidence is given a positive weight, and disconfirmatory evidence is given a negative

weight.

Insert Figure 1 about here

Figure 1 provides a schematic representation of the kind of mapping network established by the ACME program. The input to the program consists of predicate-calculus representations of the source and target analogs. The abstract example in Figure 1 includes three 1-place predicates, three 2-place predicates, and three constants in each analog. The program constructs units corresponding to each mapping hypothesis consistent with the logical-compatibility constraint. (If the analogs are problems, the role-identity constraint is also used to further restrict the units that are established.) Thus for each 2-place predicate in the source, for example, units are established for each possible pairing with a 2-place target predicate (i.e., $D=V$, $D=W$, and $D=X$). In addition, for each source element a unit is constructed to represent the possibility of a null map (e.g., $D=0$)—i.e., the program explicitly considers the possibility that some source elements may not map to any target elements. In addition to the units representing mapping hypotheses, there is a special “semantic unit” that represents the system’s prior assessment of the degree of semantic similarity between each pair of meaningful concepts in the source and target.

Once the units are established, links are formed between them to represent the constraints of uniqueness, relational consistency, and semantic similarity. Figure 1 illustrates a subset of the links that would be formed for the example. All links have symmetric weights. To enforce uniqueness, inhibitory links (dotted lines) connect all competing hypotheses (e.g., the unit $D=V$ inhibits $D=W$ and $E=V$). In a complete version of Figure 1, inhibitory links would connect all units in the same row or column within each subset (2-place predicate mappings, 1-place predicate

mappings, and constant mappings), with the exception of the column of units for null mappings (since any number of source elements might fail to map to the target). To enforce relational consistency, excitatory links connect those units based on the same relation. For example, the predicate-mapping hypothesis $D=V$ supports mappings between the corresponding constants, $a=s$ and $b=t$, which in turn support each other. Similarly, the unit $A=S$ supports $a=s$ and $B=T$ supports $b=t$. The semantic-similarity constraint is enforced by placing excitatory links from the semantic unit to all units representing mappings between meaningful concepts (i.e., predicates rather than constants), as well as to units for null mappings of predicates, such as $D=0$. The weights on these links are made proportional to the degree of semantic similarity between the mapped concepts. The links from the semantic unit to units representing null mappings are uniformly set at a minimal positive value.

The manner in which the network is run to arrive at a solution is a straightforward application of constraint-satisfaction methods that have been investigated extensively in other applications (see Rumelhart et al., 1986). To initialize the network, the activation level of the semantic unit is fixed at 1 and the activations of all other units are set to 0. On each cycle of activity, all units (except the semantic unit) have their activation levels updated on the basis of the activation levels and weights associated with neighboring units and links. The updating procedure is adapted from that employed in McClelland and Rumelhart's (1981) model of word recognition. The activation level of unit j on cycle t is given by

$$a_j(t+1) = a_j(t)(1 - \Theta) + \begin{cases} net_j(max - a_j(t)) & \text{if } net_j > 0 \\ net_j(a_j(t) - min) & \text{otherwise,} \end{cases}$$

where Θ is a decay parameter, $min = -1$, $max = 1$, and $net_j = \sum_i w_{ij}a_i(t)$. The degree to which the activation levels of units satisfy the constraints imposed by the weights on links is given by a measure termed G , defined¹ as

$$G(t) = \sum_i \sum_j w_{ij} a_i(t) a_j(t).$$

On each cycle the activation adjustments produce a move to a new global state of activation that increases the value of G (Hopfield, 1982). The value of G can be interpreted as a rough index of the overall fit of the emerging mapping to the constraints of uniqueness, relational consistency, and similarity.

Applications of ACME

Table 1 lists the principal analogies to which ACME has been applied, along with the number of units and links that were formed for each. Because translation of analogies in natural language into predicate calculus inputs is somewhat arbitrary, these applications do not constitute strict tests of the theory implemented in ACME. Nevertheless, they show that ACME is applicable to several different kinds of analogies and is consistent with experimental results concerning when analogical mapping is difficult for people. In all these simulations the value of Θ , the decay rate, was set at .01, the inhibitory weights for the uniqueness constraint were set at .02, the excitatory weights for relational consistency² were set at .05, and the weights from the semantic unit were allowed to range from a minimum value of .01 to a maximum of .1 for identical predicates.

Insert Table 1 about here

Problem Analogies

ACME has been applied to a number of problem analogies involving the use of a "convergence" solution, in which several weak forces are applied simultaneously to a centrally-located object

in order to achieve the effect of a single large force (Gick & Holyoak, 1980, 1983; Holyoak & Koh, 1987). In experimental work using these materials, the target analog has typically been a "radiation problem", in which a doctor must find a way to use a ray to destroy a stomach tumor without harming the surrounding tissue (Duncker, 1945). Holyoak and Koh (1987) compared the effectiveness of four alternative versions of a source analog based on a "lightbulb problem", in which a broken filament in a lightbulb must be repaired. The four versions were defined by two types of variations. Surface similarity to the ray used in the target was varied by manipulating whether the force used in the lightbulb problem was a laser (highly similar to a ray) or an ultrasound beam (less similar). Similarity of problem constraints was also varied. The similar constraint in the source was that it was necessary to avoid breaking the fragile glass bulb surrounding the filament (analogous to avoiding injury to the tissue surrounding the tumor). The dissimilar constraint was that a force of sufficiently high intensity was not available. Table 2 presents predicate-calculus representations of the "laser/fragile-glass" version of the lightbulb problem and of the radiation problem. These, together with similar representations of the other lightbulb versions, were used as inputs to ACME. In each run the possible mapping hypotheses were limited by the role-identity constraint (i.e., goal elements must map to goal elements, and so on).

Insert Table 2 about here

Holyoak and Koh (1987) measured the percent of undergraduates who produced the convergence solution to the radiation problem after reading one of the four versions of the lightbulb problem, both before a hint to use the source was given and in total after a hint was provided. For comparison with ACME's mapping results, these data are provided at the bottom of Table 3.

Since ACME is modeling mapping only, not retrieval, the more relevant comparison is with the number of solutions after a hint was given. Table 3 presents the results of running ACME on four versions of the convergence problem. Two global measures of the difficulty of the mapping are reported: the value of G after ten cycles (the higher the value, the greater the degree of fit to the constraints), and "cycles to success", the number of cycles required for all of the correct individual mapping hypotheses to reach an activation level exceeding that of their nearest competitor. ACME is able to find the correct set of mappings in all four cases. G is substantially lower and cycles to success is substantially higher in the two bad-constraint conditions, which are just the ones that people have most trouble with. ACME shows a very slight effect of surface similarity, on G only, which arises from the differential weights from the semantic unit to the more similar predicate pair *laser-ray* (set at .05) as compared to the less similar pair *ultrasound-ray* (.01). As the data from the Holyoak and Koh study indicate, people are also able to derive the mapping equally well in the *laser* and *ultrasound* conditions once a hint is provided.

Insert Table 3 and 4 about here

The activation levels after 10 cycles of the winning hypotheses in the first condition are shown in Table 4. Note that there is a considerable range of activation values, reflecting variations in the degree of support for different mapping hypotheses.

ACME has also been applied to another convergence analogy, in which the target is the radiation problem and the source is the "fortress problem" used by Gick and Holyoak (1980). In the latter problem a general divides his army into small groups and has them converge simultaneously on a fortress to capture it. Relative to all the versions of the lightbulb problem, the concepts in the fortress problem are less similar to those in the radiation problem. Accordingly,

mapping the fortress and radiation problems requires more cycles to success (14) than mapping the basic laser and radiation problems, and produces a lower *G* value, .10, after 10 cycles. Human problem solvers also find it harder to use the fortress problem as opposed to any of the lightbulb problems as a source analog for the radiation problem. We advise, however, that comparing *G* values across problems is problematic because of its sensitivity to representation changes.

ACME's performance on the various convergence analogies illustrates one of its major strengths relative to the SME program of Falkenhainer et al. (1986). SME would be unable to capture more than isolated fragments of these analogies because it is restricted to finding mappings between identical relations. Although the convergence analogs include mappings between some identical relations (e.g., *surround-surround*), other mappings are between relations that have minimal similarity (e.g., *fuse-destroy*). ACME can identify such mappings by exploiting the constraints of relational consistency and uniqueness. We will provide further demonstrations below of ACME's performance with semantically-dissimilar relations.

Explanatory Analogies

ACME has been applied to the explanatory analogies discussed by Gentner (1983, forthcoming; Falkenhainer et al., 1986): the analogy between the flow of water caused by differential pressure and the flow of heat caused by differential temperature; and the analogy between the motion of planets around the sun and of electrons around an atomic nucleus. These examples allow a close comparison of the ACME and SME programs, since Falkenhainer et al. (1986) describe the representations used as input to SME in sufficient detail that we could provide essentially the same information about each analog to ACME.

Table 5 presents predicate-calculus representations of the water-flow/heat-flow analogy that were used as inputs to ACME. These representations differ in certain respects from those used

by Falkenhainer et al. in testing SME. The representations used by Falkenhainer et al. treat the predicate "cause" as a second-order predicate with propositions as arguments. But causality is usually analyzed not as a relation between predicates or propositions, but as a relation between events or objects (Davidson, 1980; Steiner, 1986). Accordingly, our representations introduce constants representing events as a final argument of relations. For example, we interpret the temperature of coffee being greater than that of the ice cube as part of an event which can cause other events. Our representations also differ from those of Falkenhainer et al. in treating functions such as temperature in the way of standard logic, as 2-place relations between an object and a value (Mendelson, 1964, p. 7), rather than in the computer-language way as operators that return a value. Despite these representational differences, our first-order predicate-calculus representations encode essentially the same information as that which Falkenhainer et al. provided to the SME program.

Insert Tables 5 and 6 about here

Table 6 presents the activation levels of selected mapping hypotheses after 10 cycles. The complete correct mapping emerges after 7 cycles. There are two major impediments to a successful map from water flow to heat flow. First, there is the misleading information that both water and coffee are liquids and have a flat top. Although ACME initially maps water to coffee, relational information encoded in the network quickly enables it to provide higher activation to the unit representing the hypothesis that maps water to heat. As the values in Table 6 indicate, the mapping from water to heat emerges as a clear victor over the alternative possibility of mapping water to coffee. Also note that the source predicate *clear* is correctly mapped onto the null element.

The second major impediment to a successful map is the irrelevant information concerning the diameters of the beaker and vial which would encourage the map of diameter to pressure as an equal competitor to the correct map of temperature to pressure. SME selects the correct map on the basis of Gentner's principle of systematicity, which selects predicates related by higher-order predicates. In contrast, we view the preferability of the temperature-pressure map as largely a pragmatic matter of the intended use of the the analogy. If water flow is being used to explain heat flow, than aspects of water systems that affect its flow (pressure differences rather than diameter differences) should be selected for transfer.

In the representation of the heat-flow analogy in Table 5, the information-seeking purpose of the analogy is captured by the proposition (*cause (?event? event17)*) in the heat-flow representation, where the dummy constant "?event?" represents the unknown cause of heat flow (event17). This dummy constant signals that a purpose of the mapping is to identify an actual constant in the heat situation that can fill the empty argument slot. ACME treats such constants specially, constructing weak excitatory links to units that have the potential of providing the desired information. Since *event6* concerning the greater temperature has this desired feature by virtue of its appearance in the proposition (*cause (event6 event7)*), whereas *event9* concerning the greater diameter does not, units for mapping the former are slightly preferred to units for mapping the latter. Through subsequent network adjustments this results in higher activation of the unit that maps temperature and pressure than of the one that maps diameter and pressure. Note that the predicate *diameter* in fact maps to *temperature* more successfully than it maps to anything else; however, *pressure* maps to *temperature* slightly more successfully still.

The role-identity constraint was not used in mapping this or any of the other remaining analogies we will consider in this paper, since these analogs lack the structure of problems. Weights from the semantic unit to predicate-mapping hypotheses were set equal to .1 for identical

predicates and to .01 otherwise.

ACME is also able to produce the appropriate mapping for the solar-system/atom analogy that Falkenhainer et al. used to test SME. Since this analogy does not provide any additional complexities (a complete mapping is found in just 2 cycles), we will not describe the results in detail. ACME's ability to find essentially the same mappings in these two examples as were obtained with SME, without invoking the systematicity principle, raises the issue of how systematicity relates to the constraints that govern ACME. We will defer discussion of this issue until we have presented additional relevant tests of ACME.

Story Analogies

Additional evidence concerning ACME's ability to account for empirical evidence relating to the effect of systematicity on mapping is provided by a study performed by Gentner and Toupin (1986). This experiment investigated the effects both of systematicity and *transparency*: the degree to which similar objects serve similar functions in the analogy. Gentner and Toupin presented two groups of children, aged 4-6 years and 8-10 years, with a series of simple stories. After the child had acted out one version of a story with props, the experimenter asked him or her to act out the same story with different characters.

Table 7 presents a simplified version of one of these stories that served as the basis for a simulation by ACME, and Table 8 presents the actual predicate-calculus representation provided to the program. As indicated in Table 7, each source story was used across children in either a "systematic" or a "nonsystematic" form. The systematic version differed from the nonsystematic version in that it added additional information relevant to the causes of events in the story (e.g., the cat's jealousy caused its anger). Transparency was varied by manipulating the similarity of the animals in the various roles. In the example used in the simulation, the target analog

involved a dog, seal, and penguin. In the S/S condition, the source analog involved similar characters playing similar roles (cat, walrus, and seagull). In the D condition, all the characters were quite different from those in the target (camel, lion, and giraffe). In the cross-mapped S/D condition, similar characters were used, but these played different roles than did the corresponding characters in the target (seagull, cat, and walrus).

Insert Tables 7 and 8 about here

Gentner and Toupin found that both systematicity and transparency affected the accuracy with which children enacted the target stories. The two effects interacted, in that performance was uniformly good, regardless of systematicity, when similar characters played similar roles (S/S condition). As the transparency of the mapping decreased from the S/S to the D and the S/D conditions, performance declined, and the advantage of the systematic over the unsystematic version increased. The positive impact of systematicity was more pronounced for the older group of children.

In order to simulate these results, predicate-calculus representations of the stories were used as inputs to ACME (see Table 8). If the similarity of the characters in the source and target was high, the similarity weight for the corresponding predicate-mapping unit was set at .03; if the similarity was low, the weight was set at .01. Units for pairings of identical predicates were given similarity weights of .1. Table 9 presents global measures of the ease of mapping in each of the six conditions. Values of cycles to success correspond fairly well to the degree of difficulty that Gentner and Toupin's subjects had with analogies in the different conditions: the mapping is increasingly difficult to derive as either systematicity or transparency decreases, although ACME shows no interaction effect. Unlike the case for the simulations of convergence

analogies, however, G here does not correspond well to ease of solution. The greater activation present in the S/D condition as the result of semantic similarity leads to a relatively high G , even though the mismatch between characters and roles makes the problem hard both for subjects and for ACME with respect to cycles to success. Because the excitatory weights exceed the inhibitory weights, ACME manages to satisfy both the similarity constraints and the relational-consistency constraints to some degree in the S/D condition, despite the fact that these conflict. G is therefore kept relatively high, even though the correct mapping emerges slowly.

Insert Table 9 about here

It is clear why ACME's cycles to success are sensitive to both transparency and systematicity. With respect to transparency, in the S/S condition the similarity constraint and the relational-consistency constraint are in agreement, and cooperate to produce the appropriate mappings of characters. In contrast, in the S/D condition the two factors are in direct competition, making the mapping much more difficult to discover. The D condition, in which similarity carries no weight, and hence neither helps nor hinders relational consistency, is intermediate in difficulty.

The systematicity factor in the Gentner and Toupin study is directly correlated with relational consistency. Given that the subjects' task always involved mapping an isomorphic source and target, adding additional causal structure to the source necessarily increased the number of mappable predicates linking the source and target, which in turn increased overall relational consistency. These results, like the simulation of the explanatory analogies of Falkenhainer et al. (1986), indicate that the relational-consistency constraint can perform much, if not all, of the theoretical work that structure-mapping theory ascribes to systematicity.

As we noted above, Gentner and Toupin found that the younger children benefited less from high systematicity than did the older children. The authors suggested that focus on systematicity increases with age. In terms of the present theory, it is possible that with age children learn to place greater weight on relational consistency, and less on the similarity constraint. It is also possible, however, that the younger children in the Gentner and Toupin (1986) study simply failed to grasp some of the causal structure provided in the systematic stories, and hence encoded the source stories imperfectly. Thus the lesser benefit they derived from the systematic versions need not imply insensitivity to the relational-consistency constraint.

A Formal Isomorphism

As we pointed out in connection with the simulation of problem analogies, ACME is able to use relational-consistency information to map predicates that are not semantically identical, or even similar. In fact, if two analogs are isomorphic, it should be possible to derive an appropriate mapping even in the complete absence of information about semantic similarities. Table 10 presents a formal analogy between addition of numbers and union of sets that was used to demonstrate this point. Both addition and union have the abstract mathematical properties of commutativity, associativity, and the existence of an identity element (0 for numbers, the empty set Φ for sets). ACME was given predicate-calculus representations of these two analogs, with no identical elements (note that number-equality and set-equality are given distinct symbols), and with all semantic weights set equal to the minimal value. Thus only weights based on relational consistency and uniqueness, coupled with the logical-compatibility constraint, provided information about the optimal mapping.

Insert Table 10 about here

ACME quickly derives the appropriate correspondences between sum and union and between equality of numbers and equality of sets; it also succeeds in mapping the two identity elements. However, the program fails to find one of the correspondences of particular numbers with particular sets. The exception arises because the first-order predicate-calculus representation of the two analogs requires the introduction of numerous intermediate values, such as the sum of N1 and N2 and the union of S1 and S2, which get in the way of finding the complete isomorphism. (The representations given to the program did not explicitly group the components of each analog into three distinct equations.)

Although ACME's success is not complete on this problem, we note that the SME program, which is intended to be a purely syntactic theory of mapping, is incapable of even beginning to derive such formal analogies. Because SME depends on having identical top-level predicates in the source and target, it cannot recognize analogies based purely on relational structure. By contrast, although ACME is highly sensitive to semantic and pragmatic cues to the appropriate mapping when these are available, in the absence of such cues it is capable of mapping isomorphic analogs using purely internal constraints.

Metaphor

To explore the performance of ACME in metaphorical mapping, we gave the program predicate-calculus representations of the knowledge underlying a metaphor that has been analyzed in detail by Kittay (1987). The metaphor is derived from a passage in Plato's *Theaetetus* in which Socrates declares himself to be a "midwife of ideas", elaborating the metaphor at length. Table 11 contains

predicate-calculus representations based on Kittay's analysis of the source analog concerning the role of a midwife and of the target analog concerning the role of a philosopher-teacher. Roughly, Socrates claims that he is like a midwife in that he introduces the student to intellectual partners, just as a midwife often serves first as a match-maker; and he helps the student evaluate the truth or falsity of his ideas much as a midwife helps a mother to deliver a child.

Insert Table 11 about here

Even without any information about semantic similarity, ACME is able to map these two analogs correctly after 13 cycles. The sentences expressing causal relations in the two analogs are not essential here: deletion of them still allows a complete mapping to be discovered. Thus the Socratic metaphor, like the addition/union analogy, illustrates the power of the relational-consistency and uniqueness constraints.

General Discussion

Our constraint-satisfaction theory of analogical mapping, as implemented in the ACME program, applies to diverse analogies. We will now consider more closely the relationship between the theory and a major previous account of mapping, and also point out some limitations and possible extensions of the current theory.

Relationship of ACME to SME

As we pointed out earlier, no previous account of analogical mapping has integrated constraints based on semantics, pragmatics, and the syntax of predicate-argument structure. This integration has enabled ACME to deal with an unusually broad range of analogies. The most comparable

alternative is the SME program based on Gentner's structure-mapping theory (Falkenhainer et al., 1986; Gentner, 1983, forthcoming). ACME and SME have several important similarities. Most notably, both models are intended as general theories of the mapping component, both derive a global "best" mapping from a set of constituent mappings, and both emphasize the role of predicate mappings in enforcing mappings between corresponding arguments.

The differences, however, are also notable. ACME uses representations expressed in first-order predicate calculus; thus in contrast to SME, the formal "order" of predicates is not a factor used to constrain any aspect of the analogy process. ACME is primarily a model of mapping, rather than of transfer; however, the model in no way precludes transfer of attributes in addition to relations, as does SME. More generally, ACME includes semantic and pragmatic constraints on the mapping component, as well as constraints based on predicate-argument relations.

Gentner's (forthcoming) systematicity principle postulates that people prefer to map connected systems of relations governed by higher-order relations with inferential import, rather than isolated predicates. Our theory also emphasizes systems of relations, but identifies them by using the five constraints, particularly relational consistency, rather than higher-order relations. Although Gentner's systematicity and relational consistency are related, the two concepts are distinct. Systematicity is defined by the interrelatedness of predicate-argument structure *within* a single analog, whereas relational consistency is defined by the interrelatedness of predicate-argument structure *between* two analogs. Systematicity can be viewed as a prerequisite for relational consistency, in that if the two analogs lack internal structure, then *ipso facto* there is no basis for defining consistency of the mapping between the two. On the other hand, high systematicity in no way ensures relational consistency; for example, a person can have highly systematic representations of both the motion of planets and of Cuban politics, yet there is no obvious analogy between the two domains.

The SME program in fact attends to both relational consistency and to systematicity. The program first identifies subsets of consistently-mappable relations in the two analogs. "Consistency" in SME is similar to our uniqueness and relational-consistency constraints, but it is treated as a strict criterion, rather than as a matter of degree, and is restricted to mappings between identical relations. Once consistently-mappable subsets have been identified, SME evaluates the systematicity of each subset to select a preferred mapping. In contrast, ACME attends to relational consistency and uniqueness, but not in a direct way to systematicity. In fact, as demonstrated by our simulation of the empirical effect of systematicity obtained by Gentner and Toupin (1986), sensitivity to relational consistency is sufficient to ensure sensitivity to systematicity. For to the extent that the mapping between two analogs satisfies the constraint of maximizing relational consistency, the two analogs must in fact have a corresponding degree of internal systematic structure. In short, the mapping component can operate successfully by attending directly to relational consistency but not systematicity, whereas the converse is not true.

A major advantage that use of the relational-consistency and uniqueness constraints conveys on ACME is its ability to find mappings between elements that are not semantically identical, or even similar. This power is demonstrated in the partial mapping found between the formal analogs of addition of number and union of sets, in which *none* of the mapped elements are similar. It also played an important role in mapping problem analogs and metaphors in which many of the mapped predicates were semantically dissimilar. In one experiment, ACME was run on the midwife/Socrates metaphor (see Table 11) with no use of "cause" and with the predicate "helps" changed to "aids". All weights from the semantic unit were set at a uniform minimal value, so that the program had no knowledge of identical or semantically similar predicates. Nonetheless, ACME arrived at the correct mapping in fewer than 30 cycles of updating activation. Thus

sensitivity to predicate-argument structure allows the mapping component to *find* important similarities between predicates, rather than depending on the similarities being precoded in the initial representations of the analogs. This creative aspect of analogy is not well-captured by SME or other previous models of mapping that are highly dependent on preexisting similarities or identities.

A related difference between the SME and ACME programs involves the "tightness" of constraints on mapping. As noted above, SME begins by identifying consistently-mappable subsets of the analogs. Any violation of the strong constraint that mapped relations must be identical marks the limit of a consistently-mappable subset. The program typically yields several such subsets, ranked in order of "goodness". In contrast, ACME treats the constraints of relational consistency, uniqueness, and semantic similarity as "pressures" that operate in parallel to find a single mapping that best satisfies all of the converging and/or competing constraints (cf. Hofstadter, 1984). The program on any one run finds a single set of "best" mapping hypotheses (although relatively high activation levels on other units convey information about possible alternative mappings).

Extensions of the Theory

Although our constraint-satisfaction theory of analogical mapping appears powerful in its intended domain, many other important issues about analogy remain unresolved. Most notably, the model of mapping needs to be incorporated into a broader theory of all phases of analogical reasoning (Holyoak & Thagard, forthcoming). Of particular interest is the link between the initial spontaneous retrieval of plausibly useful analogs and the subsequent mapping process. There is evidence that retrieval is more heavily influenced by semantic similarity of predicates than is mapping (Gentner & Landers, 1985; Holyoak & Koh, 1987), although retrieval also seems to be

influenced by deeper forms of relational similarity (Holyoak, 1984; Holyoak & Koh, 1987; Schank, 1982). A model of the retrieval of analogs should presumably provide the initial computations of predicate similarities that are used to establish the weights from the semantic unit used by ACME.

In addition to the need to address other components of analogy, other issues related to mapping remain outstanding. In exploratory simulations using homomorphs of the well-known "missionaries and cannibals" problem, ACME has trouble dealing with one-many mappings. This shortcoming of the program very likely reflects a broad area in which theories of analogy require extension. To find useful analogies between complex analogs, it will often be necessary to interweave the mapping component with strategic manipulation of the representations of the source and target. Non-isomorphic correspondences may be found if it is possible to tentatively group elements of each analog into sets, which can then be treated as unitary objects. More generally, it may often be advantageous to attempt mappings at different levels of abstraction. Thus although it has been useful to model analogical mapping as a separate component, future theoretical development will likely require that mapping be treated in a more integrated way with other aspects of analogy and general reasoning.

Finally, we note that the general form of the theory we have proposed for analogical mapping—a set of constraints satisfiable via a cooperative algorithm—may well be applicable to other high-level cognitive processes. Lehnert (1987), for example, describes a sentence analyzer that uses a constraint network to parse sentences into case-frame meaning relationships. The parallelism of human information processing, which is so evident in lower-level perception and memory retrieval, may extend to important aspects of reasoning and problem solving as well.

Footnotes

The research reported in this paper was supported by Contract MDA903-86-K-0297 from the Army Research Institute. The work was presented at the meeting of the Society for Philosophy and Psychology held at the University of California, San Diego, on June 21, 1987. We thank Laura Novick for helpful comments on an earlier draft.

1. The formula for G used in ACME is a simpler form of that used elsewhere (Rumelhart et al., 1986), in that the present algorithm operates only on internal weights, and does not involve any external inputs.

2. The relational-consistency weight is incremented for each cooccurrence of two mappings hypotheses. Thus if two relations each support both $A=R$ and $B=S$, the excitatory link between these two units would receive double the basic weight. Such multiple cooccurrences occurred in the simulation of the addition/union analogy.

References

- Black, M. (1962). *Models and metaphors*. Ithaca, N.Y.: Cornell University Press.
- Brown, A. L., Kane, M. J., & Echols, K. (1986). Young children's mental models determine transfer across problems with a common goal structure. *Cognitive Development*, 1, 103-122.
- Burstein, M. H. (1986). A model of learning by incremental analogical reasoning and debugging. In R. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach*, Vol. 2. Los Altos, Calif.: Kaufmann.
- Carbonell, J. G. (1983). Learning by analogy: Formulating and generalizing plans from past experience. In R. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach*. Palo Alto, Calif.: Tioga Press.
- Carbonell, J. G. (1986). Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In R. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach*, Vol. 2. Los Altos, Calif.: Kaufmann.
- Darden, L. (1983). Artificial intelligence and philosophy of science: Reasoning by analogy in theory construction. In P. Asquith & T. Nickles (Eds.), *PSA 1982*, Vol. 2. East Lansing, Mich.: Philosophy of Science Society.
- Davidson, D. (1980). *Essay on action and events*. Oxford: Oxford University Press.
- Duncker, K. (1945). *On problem solving*. *Psychological Monographs*, 58 (Whole No. 270).
- Falkenhainer, B., Forbus, K. D., & Gentner, D. (1986). The structure-mapping engine. *Proceedings of the Meeting of the American Association for Artificial Intelligence*. Los Altos, Calif.: Kaufmann.
- Gentner, D. (1982). Are scientific analogies metaphors? In D. S. Miall (Ed.), *Metaphor: Problems and perspectives*. Brighton, England: Harvester Press.
- Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7, 155-170.

- Gentner, D. (forthcoming). The mechanisms of analogical reasoning. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning*.
- Gentner, D., & Landers, R. (1985, November). Analogical access: A good match is hard to find. Paper presented at the annual meeting of the Psychonomic Society, Boston.
- Gentner, D., & Toupin, C. (1986). Systematicity and surface similarity in the development of analogy. *Cognitive Science*, 10, 277-300.
- Gick, M. L., & Holyoak, K. J. (1980). Analogical problem solving. *Cognitive Psychology*, 12, 306-355.
- Gick, M. L., & Holyoak, K. J. (1980). Schema induction and analogical transfer. *Cognitive Psychology*, 15, 1-38.
- Hall, R. (1986). Understanding analogical reasoning: Computational approaches. Irvine Computational Intelligence Project Technical Report 86-11. Irvine, Calif.: University of California at Irvine Department of Information and Computer Science.
- Hofstadter, D. R. (1984). The Copycat project: An experiment in nondeterministic and creative analogies. Cambridge, Mass.: MIT A.I. Laboratory Memo 755.
- Hesse, M. B. (1966). *Models and analogies in science*. Notre Dame, Ind.: Notre Dame University Press.
- Holland, J. H., Holyoak, K. J., Nisbett, R. E., & Thagard, P. (1986). *Induction: Processes of inference, learning, and discovery*. Cambridge, Mass.: Bradford Books/MIT Press.
- Holyoak, K. J. (1982). An analogical framework for literary interpretation. *Poetics*, 11, 105-126.
- Holyoak, K. J. (1984). Analogical thinking and human intelligence. In R. J. Sternberg (Ed.), *Advances in the psychology of human intelligence*, Vol. 2. Hillsdale, N.J.: Erlbaum.
- Holyoak, K. J. (1985). The pragmatics of analogical transfer. In G. H. Bower (Ed.), *The psychology of learning and motivation*, Vol. 19. New York: Academic Press.
- Holyoak, K. J., & Koh, K. (1987). Surface and structural similarity in analogical transfer. *Memory &*

Cognition.

- Holyoak, K. J., Junn, E. N., & Billman, D. O. (1984). Development of analogical problem-solving skill. *Child Development*, 55, 2042-2055.
- Holyoak, K. J., & Thagard, P. (forthcoming). A computational model of analogical problem solving. In S. Vosniadou & A. Ortony (Ed.), *Similarity and analogical reasoning*.
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences, USA*, 79, 2554-2558.
- Kedar-Cabelli, S. (1985). Purpose-directed analogy. *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*.
- Kittay, E. (1987). *Metaphor: Its cognitive force and linguistic structure*. Oxford: Clarendon Press.
- Lehnert, W. (1987). Learning to integrate syntax and semantics. In P. Langley (Ed.), *Proceedings of the Fourth International Workshop on Machine Learning*. Los Altos, Calif.: Kaufmann.
- Marr, D. (1982). *Vision*. New York: Freeman.
- Marr, D., & Poggio, T. (1976). Cooperative computation of stereo disparity. *Science*, 194, 283-287.
- Mendelson, E. (1964). *Introduction to mathematical logic*. New York: Van Nostrand Reinhold.
- Miller, G. A. (1979). Images and models, similes and metaphors. In A. Ortony (Ed.), *Metaphor and thought*. Cambridge, England: Cambridge University Press.
- McClelland, J. L., & Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception: Part 1. An account of basic findings. *Psychological Review*, 88, 375-407.
- Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, N.J.: Prentice-Hall.
- Novick, L. R. (1986). *Analogical transfer in expert and novice problem solvers*. Unpublished doctoral dissertation, Stanford University.
- Polya, G. (1973). *Mathematics and plausible reasoning*, Vol. 1. Princeton, N.J.: Princeton University Press.

- Ross, B. H. (in press). This is like that: The use of earlier problems and the separation of similarity effects. *Journal of Experimental Psychology: Learning, Memory, and Cognition*.
- Rumelhart, D. E., Smolensky, P., McClelland, J. L., & Hinton, G. E. (1986). Schemata and sequential thought processes in PDP models. In J. L. McClelland, D. E. Rumelhart, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition*, Vol. 2. Cambridge, Mass.: Bradford Books/MIT Press.
- Schank, R. C. (1982). *Dynamic memory*. Cambridge, Mass.: Cambridge University Press.
- Steiner, M. (1986). Events and causality. *Journal of Philosophy*, 83, 249-264.
- Thagard, P. (1984). Frames, knowledge, and inference. *Synthese*, 61, 301-318.
- Thagard, P. (1986). Parallel computation and the mind-body problem. *Cognitive Science*, 10, 301-318.
- Thagard, P. (in press, a). Dimensions of analogy. In D. Helman (Ed.), *Analogical reasoning*. Dordrecht, Holland: Reidel.
- Thagard, P. (in press, b). *Computational philosophy of science*. Cambridge, Mass.: Bradford Books/MIT Press.
- Tversky, A. (1977). Features of similarity. *Psychological Review*, 84, 327-352.
- Winston, P. H. (1980). Learning and reasoning by analogy. *Communications of the ACM*, 23, 689-703.

Figure Caption

Figure 1. A schematic example of an ACME mapping network. Capital letters represent predicates, small letters represent constants. Solid lines represent excitatory connections, dotted lines represent inhibitory connections. See text for further explanation.

Appendix

Table 1

Summary of Applications of ACME

Analogs	Number of Units	Number of Symmetric Links
lightbulb/radiation problems (4 versions) (Holyoak & Koh, 1987)	100-118	853-912
fortress/radiation problems (Gick & Holyoak, 1980)	52	530
water-flow/heat-flow (Falkenhainer et al., 1986)	187	2148
solar-system/atom (Falkenhainer et al., 1986)	213	2716
jealous-animal stories (6 versions) (Gentner & Toupin, 1986)	138-217	1240-2589
midwife/Socrates (Kittay, 1987)	114	814
addition/union	186	2490

Table 2

Predicate-Calculus Representations of Lightbulb Problem
(Laser/Fragile-Glass Version) and Radiation Problem

LIGHTBULB PROBLEM (source)

Start: (laser (obj-laser))
 (bulb (obj-bulb))
 (filament (obj-filament))
 (surround (obj-bulb obj-filament))
 (outside (obj-laser obj-bulb))
 (can-produce (obj-laser obj-beams-high))
 (high-intensity (obj-beams-high))
 (can-~~destroy~~^{fuse} (obj-beams-high obj-filament))
 (can-~~destroy~~^{fuse} (obj-beams-high obj-bulb))
 (can-produce (obj-laser obj-beams-low))
 (low-intensity (obj-beams-low))
 (cannot-~~destroy~~^{fuse} (obj-beams-low obj-filament))
 (cannot-~~destroy~~^{fuse} (obj-beams-low obj-bulb))

Goals: (fuse (obj-laser obj-filament))
 (intact (obj-bulb))

RADIATION PROBLEM (target)

Start: (ray-source (obj-ray))
 (tissue (obj-tissue))
 (tumor (obj-tumor))
 (surround (obj-tissue obj-tumor))
 (outside (obj-ray obj-tissue))
 (can-produce (obj-ray obj-rays-high))
 (high-intensity (obj-rays-high))
 (can-destroy (obj-rays-high obj-tumor))
 (can-destroy (obj-rays-high obj-tissue))
 (can-produce (obj-ray obj-rays-low))
 (low-intensity (obj-rays-low))
 (cannot-destroy (obj-rays-low obj-tumor))
 (cannot-destroy (obj-rays-low obj-tissue))

Goals: (destroy (obj-ray obj-tumor))
 (alive (obj-tissue))

Table 3

Mappings Obtained for Four Versions of
Lightbulb/Radiation Problem Analogy

	Version			
	Laser/ Fragile-glass	Ultrasound/ Fragile-glass	Laser/ Insufficient- intensity	Ultrasound/ Insufficient- intensity
Cycles to Success	4	4	8	9
G after 10 cycles	5.43	5.36	0.31	0.19
Percent Convergence Solutions Prior to Hint*	69	38	33	13
Percent Convergence Solutions with Hint*	75	81	60	47

*Data from Holyoak and Koh (1987)

Table 4

Activation Values of Best Mappings of Lightbulb Problem to Radiation
Problem (Laser/Fragile-Glass Version) after 10 cycles

Unit:	Activation
laser=ray-source	0.45
filament=tumor	0.18
bulb=tissue	0.18
surround=surround	0.71
outside=outside	0.70
can-produce=can-produce	0.83
high-intensity=high-intensity	0.67
can-destroy=can-destroy	0.86
low-intensity=low-intensity	0.67
cannot-destroy=cannot-destroy	0.83
fuse=destroy	0.30
intact=alive	0.28
obj_laser=obj_ray	0.75
obj_filament=obj_tumor	0.78
obj_bulb=obj_tissue	0.79
obj_beams_high=obj_rays_high	0.82
obj_beams_low=obj_rays_low	0.78

Table 5

Predicate-Calculus Representations of Water-Flow
and Heat-Flow Analogs

WATER-FLOW (source)

```
(liquid (obj_water))
(flat-top (obj_water))
(clear (obj_beaker))

; the diameter of obj_beaker is obj_val1, as part of event1
(diameter (obj_beaker obj_val1 event1))*
(diameter (obj_vial obj_val2 event2))
(greater (obj_val1 obj_val2 event3))
(pressure (obj_beaker obj_val3 event4))
(pressure (obj_vial obj_val4 event5))
(greater (obj_val3 obj_val4 event6))

; flow: from x to y of w via z
(flow (obj_beaker obj_vial obj_water obj_pipe event7))

; pressure difference causes flow:
(cause (event6 event7))
```

HEAT-FLOW (target)

```
(liquid (obj_coffee))
(flat-top (obj_coffee))

(temperature (obj_coffee obj_val13 event14))
(temperature (obj_ice_cube obj_val14 event15))
(greater (obj_val13 obj_val14 event16))

(flow (obj_coffee obj_ice_cube obj_heat obj_bar event17))

(cause (?event? event17))
```

Note: Explanatory comments follow ";".

Table 6

Activation Values of Best Mappings of Water-Flow
to Heat-Flow after 10 cycles

Unit	Activation
liquid=liquid	0.63
flat-top=flat-top	0.64
clear=null	0.13
diameter=temperature	0.37
pressure=temperature	0.40
greater=greater	0.73
flow=flow	0.75
cause=cause	0.66
obj-pipe=obj-bar	0.47
obj-water=obj-heat	0.45
*obj_water=obj_coffee	0.30
obj-vial=obj-ice-cube	0.58
obj-beaker=obj-coffee	0.55

*Erroneous mapping hypothesis, provided for comparison purposes.

Table 7

Precis of a "Jealous Animal" Story as Used in ACME
Simulation, in Systematic and Nonsystematic Versions

The cat was jealous.

(Nonsystematic version: The cat was strong.)

The cat was friends with a walrus.

The walrus played with a seagull.

The cat was angry.

(Systematic version: Because the cat was jealous
and the walrus played with the seagull, the cat was angry.)

The cat was reckless.

(Systematic version: Because the cat was angry, it was reckless.)

The cat got in danger.

(Systematic version: Because the cat was reckless, it got in danger.)

The seagull saved the cat.

(Systematic version: Because the seagull saved the cat, the cat was friends
with the seagull.)

Table 8

Predicate-calculus Representation of a "Jealous Animal" Story:
Similar Objects/Similar Roles (Systematic Version)

```
(cat (obj_cat))
((jealous (obj_cat event16))
(walrus (obj_walrus))
(seagull (obj_seagull))
(friends (obj_cat obj_walrus))
(played (obj_walrus obj_seagull event11))
(angry (obj_cat event12))
(reckless (obj_cat event13))
(endangered (obj_cat event14))
(save (obj_seagull obj_cat event15))
(befriend (obj_cat obj_seagull event18))
(cause (event11 event12))
(cause (event12 event13))
(cause (event14 event15))
(cause (event13 event14))
(conjoin-event (event16 event12 event17))*
(cause (event17 event13))
(cause (event15 event18))
```

*The interpretation of "conjoin-event" is that two events are conjoined to form a third event. This device serves to ensure that "cause" remains a two-place relation when the cause is a conjunctive event.

Table 9

Results of ACME Runs for Six Versions of
"Jealous Animal" Stories

Versions	Cycles to Success	G after 10 cycles
Systematic:		
S/S	1	9.9
D	4	9.8
S/D	16	9.8
Nonsystematic:		
S/S	1	2.2
D	7	2.1
S/D	18	2.0

Table 10

Formal Isomorphism Between Addition of Numbers
and Union of Sets

Property	Addition	Union
commutativity:	$N_1 + N_2 = N_2 + N_1$	$S_1 \cup S_2 \equiv S_2 \cup S_1$
associativity:	$N_3 + (N_4 + N_5) = (N_3 + N_4) + N_5$	$S_3 \cup [S_4 \cup S_5] \equiv [S_3 \cup S_4] \cup S_5$
identity:	$N_6 + 0 = N_6$	$S_6 \cup \emptyset \equiv S_6$

Table 11

Predicate-Calculus Representations of Knowledge Underlying
the Metaphor "Socrates is a Midwife of Ideas"

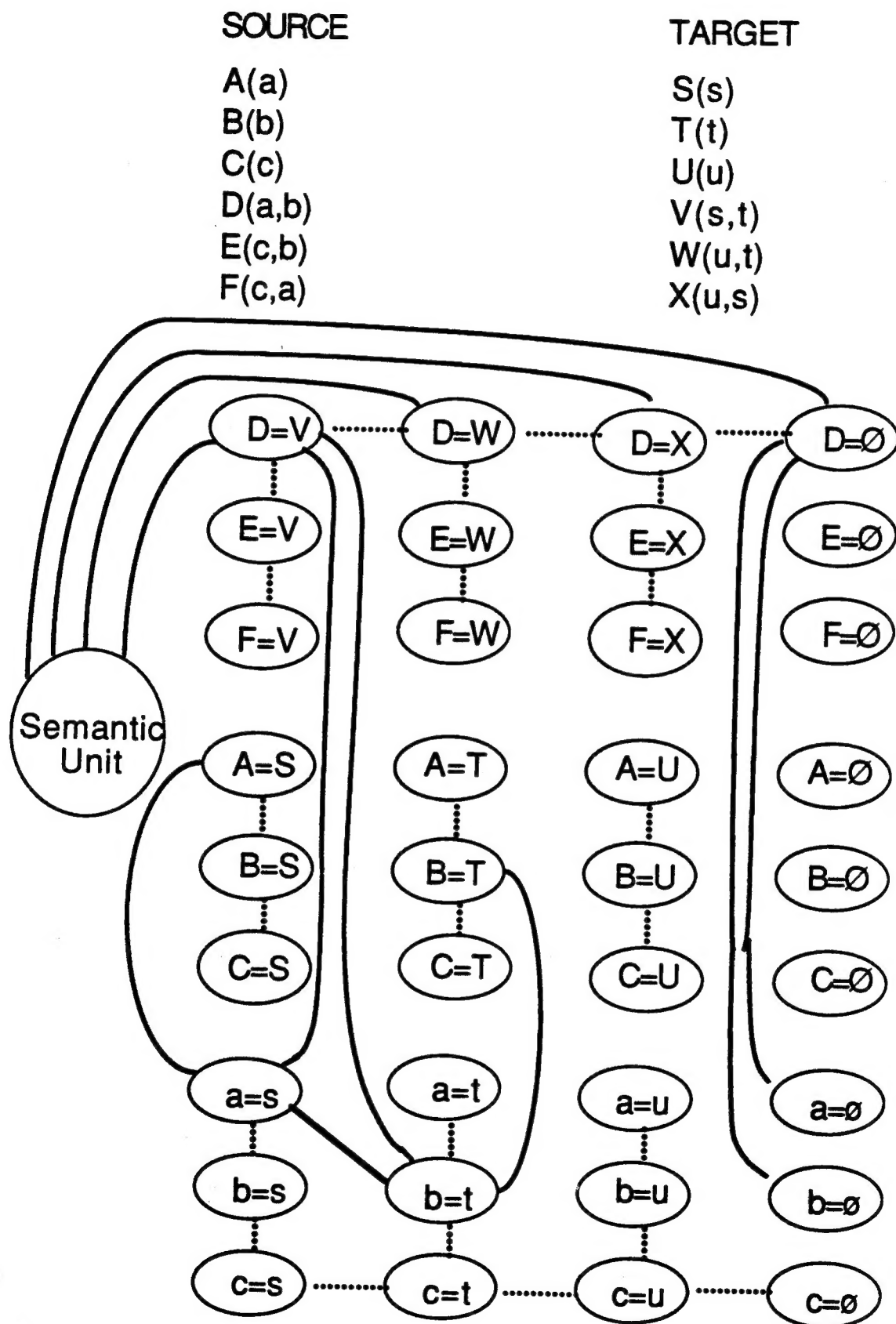
MIDWIFE (source)

```
(midwife (obj_midwife))
(mother (obj_mother))
(father (obj_father))
(child (obj_child))
(matches (obj_midwife obj_mother obj_father event11))
(conceives (obj_mother obj_child event12))
(cause (event11 event12))
(in_labor_with (obj_mother obj_child))
(helps (obj_midwife obj_mother event13))
(give_birth_to (obj_mother obj_child event14))
(cause (event13 event14))
```

SOCRATES (target)

```
(philosopher (socrates))
(student (obj_student))
(intellectual_partner (obj_partner))
(idea (obj_idea))
(introduces (socrates obj_student obj_partner event1))
(formulates (obj_student obj_idea event2))
(cause (event1 event2))
(thinks_about (obj_student obj_idea))
(tests_truth (obj_student obj_idea))
(helps (socrates obj_student event3))
(knows_truth_or_falsity (obj_student obj_idea event4))
(cause (event3 event4))
```

Setting Up a Mapping Network



ANNUAL REPORT #1

September 1, 1987 - August 31, 1987

Contract MDA903-86-K-0297

September 1, 1986 - August 31, 1989

A COGNITIVE ARCHITECTURE FOR SOLVING ILL-DEFINED PROBLEMS

University of California, Los Angeles

Princeton University

Keith J. Holyoak, Project Director

Paul R. Thagard, Co-Director

Army Research Institute for the
Behavioral and Social Sciences